

# Convex Multilinear Estimation and Operatorial Representations

Marco Signoretto

with Lieven De Lathauwer and Johan A. K. Suykens

K.U. Leuven, ESAT-SCD, Kasteelpark Arenberg 10, B-3001 Leuven (Belgium)

[marco.signoretto@esat.kuleuven.be](mailto:marco.signoretto@esat.kuleuven.be)

NIPS Workshop: Tensors, Kernels and Machine Learning

December 10, 2010

# Outline

## Convex Multilinear Estimation

- Preliminaries

- Learning with Sparsifying Penalties: from Vectors to Tensors

- CMLE: A First Order Algorithm

## Learning the Tensor with CMLE

- Unsupervised Learning

- Supervised Learning

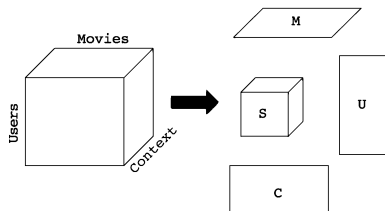
## Beyond Linearity: Tensors and Kernels

- A Framework for Non-parametric Tensor-based Models

- Experiments

# Tensor-based Models: A Machine Learning Perspective

- Tensors are tricky to deal with: non-convex problems
- + Exploit the intrinsic structure of data: multiple views
- Leads to linear models: limited discriminative power!
- + Very effective for small sample problems

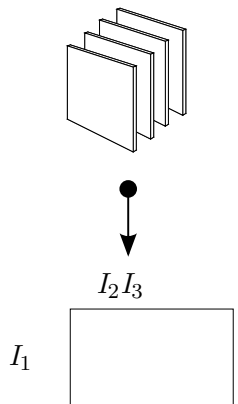


*One could argue that the most important aspect of intelligence [...] is the ability to learn from very few examples. [...] Evolution has probably done a part of the learning but so have we, when we choose for any given task an **appropriate input representation** for our learning machines.*

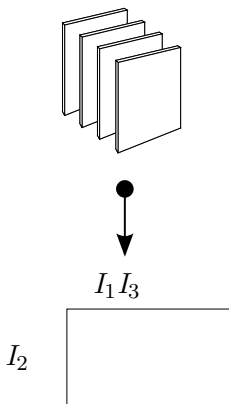
*The Mathematics of Learning: Dealing with Data*  
T. Poggio and S. Smale

## $n$ -mode Unfolding $\mathcal{A}_{<n>}$ of a Tensor $\mathcal{A}$

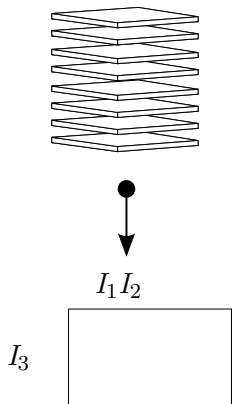
$\mathcal{A} \in \mathbb{R}^{I_1} \otimes \mathbb{R}^{I_2} \otimes \mathbb{R}^{I_3}$ . Unfolding = concatenation of slices



$$\mathcal{A}_{<1>} \in \mathbb{R}^{I_1} \otimes \mathbb{R}^{I_2 I_3}$$



$$\mathcal{A}_{<2>} \in \mathbb{R}^{I_2} \otimes \mathbb{R}^{I_1 I_3}$$



$$\mathcal{A}_{<3>} \in \mathbb{R}^{I_3} \otimes \mathbb{R}^{I_1 I_2}$$

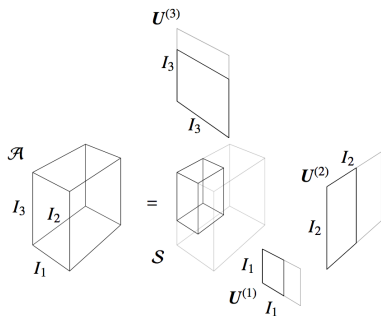
# Multilinear Singular Value Decomposition (MLSVD)

**Matrix SVD:**  $A = \sum_{i=1}^{\min\{I_1, I_2\}} \sigma_i U_i V_i^\top = \mathbf{U} \mathbf{S} \mathbf{V}^\top = \mathbf{S} \times_1 \mathbf{U} \times_2 \mathbf{V}$

Higher Order Extension: the Third Order Case

$$\mathcal{A} = \mathcal{S} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times_3 \mathbf{U}^{(3)}$$

- ▶  $\mathcal{S}$  is the core tensor (same size as  $\mathcal{A}$ )
- ▶  $\mathbf{U}^{(1)}, \mathbf{U}^{(2)}, \mathbf{U}^{(3)}$  orthogonal matrices
- ▶  $\equiv$  SVD's of the  $n$ -mode unfoldings
- ▶ notion of multilinear ranks



Truncated MLSVD leads to low multilinear ranks approximation

# A Convex Approach to Tensor-based Data Analysis

## Goal

Develop a general approach to tensor-based modeling via convex optimization

We go through three steps:

1. Sparsity Inducing Penalties: from Vectors to Higher Order Tensors
2. General Convex Formulation
3. Viable Algorithm

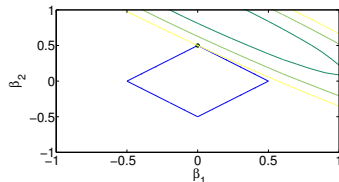
# Learning by Penalized Empirical Risk Minimization

$$\hat{W} = \arg \min_{W \in \mathbb{R}^I} \{J(W) + \lambda r(W)\}$$

- ▶  $J(W)$  measures model fit to the data (depends on the task)
- ▶  $r(W)$  is a penalty function

Sparsity inducing penalty:  $l_1$  Norm of  $W \in \mathbb{R}^I$

$$r(W) = \|W\|_1 = \sum_{i=1}^I |w_i|$$



- ▶ Best convex approximation of  $\text{card}(W) = \#\{i : w_i \neq 0\}$

# Nuclear Norms: from Matrices to Tensors

$$r(\mathbf{W}) = \|\mathbf{W}\|_1 = \sum_{i=1}^{\min\{I_1, I_2\}} \sigma_i = \|\sigma\|_1, \quad \mathbf{W} = \sum_{i=1}^{\min\{I_1, I_2\}} \sigma_i U_i V_i^\top.$$

- Best convex approximation of  $\text{rank}(\mathbf{W}) = \#\{i : \sigma_i \neq 0\}$

## Higher Order Generalization of the Nuclear Norm

Matrices

Tensors of Order N

→

$$\|\mathbf{W}\|_1 = \sum_{i=1}^{\min\{I_1, I_2\}} \sigma_i$$

$$\|\mathcal{W}\|_{1,1} = \frac{1}{N} \sum_{n=1}^N \|\mathcal{W}_{<n>}\|_1$$

$\|\mathcal{W}\|_{1,1}$  penalizes high multi-linear ranks (related to the MLSSVD)

# A class of Optimization Problems Involving Nuclear Norms

$$\begin{aligned} \min_{\mathcal{W} \in \mathbb{R}^{I_1} \otimes \mathbb{R}^{I_2} \otimes \dots \otimes \mathbb{R}^{I_N}} \quad & J(\mathcal{W}) + \lambda \|\mathcal{W}\|_{1,1} \\ \text{subject to} \quad & A(\mathcal{W}) = \mathcal{Z} \end{aligned}$$

- ▶ convex and smooth function  $J : \mathbb{R}^{I_1} \otimes \mathbb{R}^{I_2} \otimes \dots \otimes \mathbb{R}^{I_N} \rightarrow \mathbb{R}$
- ▶  $\lambda > 0$  is a user-defined regularization parameter
- ▶  $A$  is a linear transformation

Different choices of  $J$  and  $A$  give rise to different problems

# Convex MultiLinear Estimation (CMLE) Algorithm

**input**  $\mathcal{W}^{(0)}$ , dual variables  $\mathbf{P}_{(1)}^{(0)}, \dots, \mathbf{P}_{(N)}^{(0)}, \mu, c$

$k \leftarrow 1$

$\mathcal{W}^{(k)} \leftarrow \mathcal{W}^{(0)}$

$\mathbf{P}_{(n)}^{(k)} \leftarrow \mathbf{P}_{(n)}^{(0)}$

**repeat**

$\mathcal{S} \leftarrow \nabla J(\mathcal{W}^{(k)}) \leftarrow$  gradient of the problem-dependent part

**for**  $n \in \mathbb{N}_N$  **do**

$\mathbf{W}_{(n)}^{(k+1)} \leftarrow D_{\frac{\mu}{c}} \left( \mathcal{W}_{<n>}^{(k)} - \frac{1}{c} \mathcal{S}_{<n>} + \frac{1}{c} \mathbf{P}_{(n)}^{(k)} \right) \leftarrow$  singular values  
soft-thresholding  
operator

$\mathbf{P}_{(n)}^{(k+1)} \leftarrow \mathbf{P}_{(n)}^{(k)} + c \left( \mathcal{W}_{<n>}^{(k)} - \mathbf{W}_{(n)}^{(k+1)} \right)$

**end**

$\mathcal{W}^{(k+1)} \leftarrow \frac{1}{N} \sum_{n \in \mathbb{N}_N} \left( \mathbf{W}_{(n)}^{(k+1)} - \frac{1}{c} \mathbf{P}_{(n)}^{(k+1)} \right)^{<n>}$

**until** convergence criterion met

Convexity implies global optimality of the solution found!

# Supervised and Unsupervised Learning with CMLE

$$\begin{aligned} \min_{\mathcal{W} \in \mathbb{R}^{I_1} \otimes \mathbb{R}^{I_2} \otimes \dots \otimes \mathbb{R}^{I_N}} \quad & J(\mathcal{W}) + \mu \|\mathcal{W}\|_{1,1} \\ \text{subject to} \quad & A(\mathcal{W}) = \mathcal{Z} \end{aligned}$$

## Some Cases of Interest

**tensor denoising** :  $\mathcal{D}$  tensor of noisy measurements  
 $J(\mathcal{W}) = \|\mathcal{W} - \mathcal{D}\|_F^2$ ,  $A$  and  $\mathcal{Z}$  not used

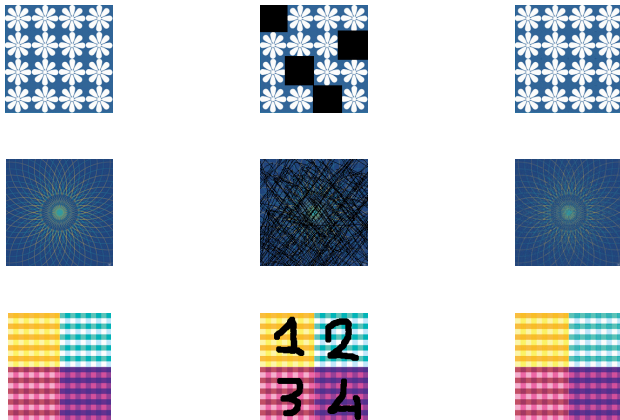
---

**tensor completion** :  $J(\mathcal{W}) = 0$   
 $A : (A(\mathcal{W}))_{j \in \mathbb{N}_m} = x_{i_1^j i_2^j \dots i_N^j}^j$ ,  $\mathcal{Z} \in \mathbb{R}^m$

---

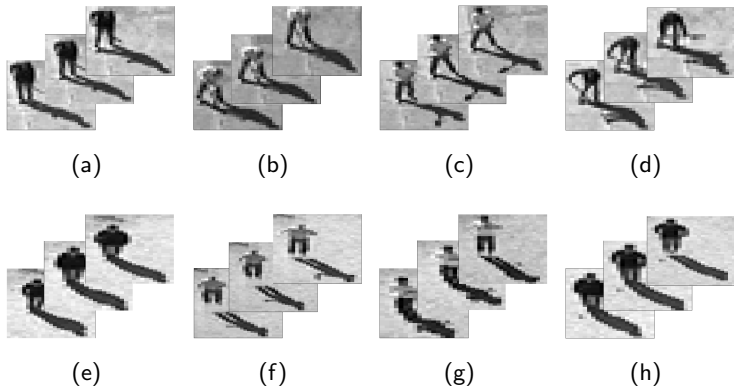
**classification / regression** : input-output pairs  $\{(\mathcal{X}^{(m)}, y_m) : m \in \mathbb{N}_M\}$   
 $J(\mathcal{W}) = \sum_{m=1}^M l(y_m, \langle \mathcal{X}^{(m)}, \mathcal{W} \rangle)$ ,  $l$  convex loss

# Tensor Completion: Image Examples



**Figure:** Original images (left column), the given pixels (middle column — black filled areas denote portion of the images that were removed) and reconstructed images (right column).

## A Supervised Case: Classification of Aerial Views



**Figure:** First 3 frames of low-resolution grayscale videos (3-way tensors). The first row ((a)-(d)) depicts input training patterns in the class *digging*, the second row ((e)-(h)) input training patterns in the class *jumping*.

## A Supervised Case: Classification of Aerial Views (cont'd)

AUC performances on test: mean (and standard deviation)

waving (I) vs waving (II)		carrying vs running	
nuc norm	lin LS-SVM	nuc norm	lin LS-SVM
<b>0.83(0.14)</b>	0.81(0.15)	<b>0.82(0.11)</b>	0.60 (0.16)
pointing vs standing		digging vs jumping	
nuc norm	lin LS-SVM	nuc norm	lin LS-SVM
<b>0.98(0.02)</b>	0.86(0.07)	<b>1(0)</b>	0.98(0.04)

**Table:** Area under the ROC curves: the tensor-based approach (nuc-norm) is compared against LS-SVMlab with linear kernel (lin LS-SVM).

# Linear Versus Non-parametric Tensor-based Models

So far..

- ▶ Nuclear norm penalty to find higher order models with low multilinear ranks
- ▶ Models were still linear in the data:

$$f(\mathcal{X}_1 + \mathcal{X}_2) = \langle \mathcal{W}, \mathcal{X}_1 + \mathcal{X}_2 \rangle = \text{vec}(\mathcal{W})^\top (\text{vec}(\mathcal{X}_1) + \text{vec}(\mathcal{X}_2))$$



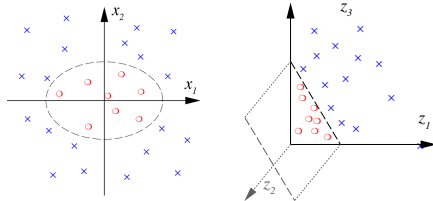
Goal

Relax linearity while exploiting the (algebraic) structure of higher order representations

# A Kernel-based Framework to Tensorial Data Analysis

## Kernel Methods

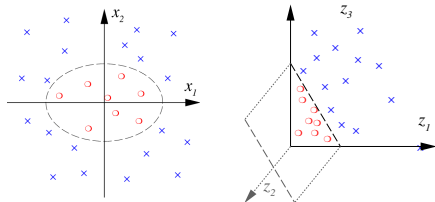
- + Lead to flexible nonlinear models
- ± Ever use specialized kernels or ignore additional structure
- What if the number of examples is very small?



# A Kernel-based Framework to Tensorial Data Analysis

## Kernel Methods

- + Lead to flexible nonlinear models
- ± Ever use specialized kernels or ignore additional structure
- What if the number of examples is very small?



## A Naïve Kernel for Tensors...

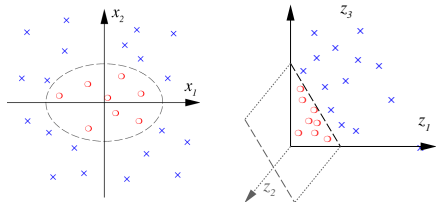
Easy to extend the Gaussian-RBF kernel to  $\mathbb{R}^{I_1} \otimes \mathbb{R}^{I_2} \otimes \dots \otimes \mathbb{R}^{I_N}$

$$k(\mathcal{X}, \mathcal{Y}) = \exp \left( -\frac{1}{2\sigma^2} \|\mathcal{X} - \mathcal{Y}\|_F^2 \right)$$

# A Kernel-based Framework to Tensorial Data Analysis

## Kernel Methods

- + Lead to flexible nonlinear models
- ± Ever use specialized kernels or ignore additional structure
- What if the number of examples is very small?



...Does not Exploit Higher Order Representations!

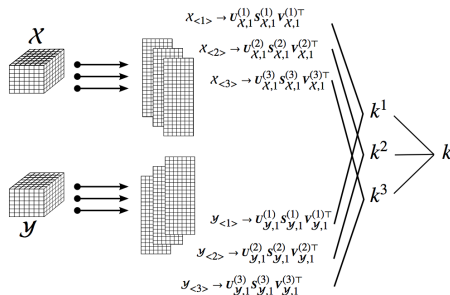
Easy to extend the Gaussian-RBF kernel to  $\mathbb{R}^{I_1} \otimes \mathbb{R}^{I_2} \otimes \dots \otimes \mathbb{R}^{I_N}$

$$k(\mathcal{X}, \mathcal{Y}) = \exp \left( -\frac{1}{2\sigma^2} \|\mathcal{X} - \mathcal{Y}\|_F^2 \right) = \exp \left( -\frac{1}{2\sigma^2} \|\text{vec}(\mathcal{X}) - \text{vec}(\mathcal{Y})\|^2 \right)$$

# A Proper Kernel Framework

- ▶ optimization approach based on primal-dual formulations
- ▶ mapping performed implicitly by a tensorial kernel

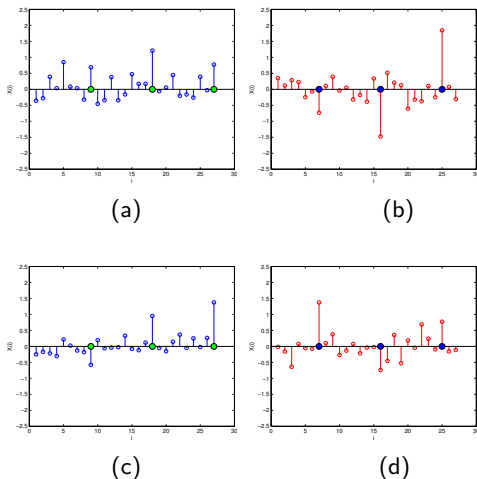
## Tensorial Kernel based on MLSSVD



Factor kernels  $k^n$  are based on *projected Frobenius norm*:

$$k^n(\mathcal{X}, \mathcal{Y}) = \exp \left( -\frac{1}{2\sigma^2} \left\| \mathbf{V}_{\mathcal{X},1}^{(n)} \mathbf{V}_{\mathcal{X},1}^{(n)\top} - \mathbf{V}_{\mathcal{Y},1}^{(n)} \mathbf{V}_{\mathcal{Y},1}^{(n)\top} \right\|_F^2 \right)$$

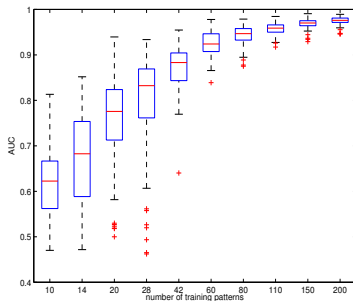
## An Illustrative Example: Classification of Sparsity Patterns



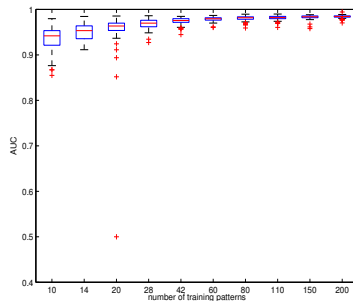
**Figure:** Classification of sparsity patterns of (noisy) vectors. Vectors in (a) and (c) belong to the first class whereas (b) and (d) belong to the second. The solid green (resp. solid blue) dots represent true non-zero entries (before noise corruption).

## An Illustrative Example: Classification of Sparsity Patterns (cont'd)

By tensor folding turned into the classification of 3-rd order tensors



(a)



(b)

**Figure:** Boxplots of AUC obtained for increasing number of training patterns for the RBF-Gaussian kernel (a) and for the tensorial kernel (b).

# Libras Dataset

- ▶ Libras = Brazilian sign language
- ▶ classes formed by 24 recordings of hand movements
- ▶ each recording is a bivariate time series (45 time instants) that is converted into  $6 \times 2 \times 40$  Hankel tensor
- ▶ classification by LS-SVM with different kernels

AUC performances on test: mean (and standard deviation)

task	tensor kernel	RBF kernel	lin kernel
class 1 VS 2	<b>0.83(0.06)</b>	0.75(0.12)	0.77(0.13)
class 1 VS 3	0.92(0.04)	<b>0.98(0.04)</b>	0.94(0.09)
class 1 VS 4	<b>1(0)</b>	0.98(0.02)	0.95(0.07)
class 1 VS 5	<b>1(0)</b>	0.97(0.06)	0.91(0.12)

# Concluding Remarks

- ▶ Convex multilinear estimation based on nuclear norm penalties
- ▶ A viable algorithm that makes use of first order information
- ▶ Applications in supervised and unsupervised learning
  
- ▶ Beyond traditional kernel methods: exploiting higher order representations

## Future Research and Open Problems

- ▶ Tensor Hunting!
- ▶ Classification of Dynamical Systems
- ▶ Nuclear Norm in a primal-dual setting?

Thank You for Your Attention!

## Selected References

- ▶ M. Signoretto, L. De Lathauwer and J. A. K. Suykens, Nuclear Norms for Tensors and Their Use for Convex Multilinear Estimation. Internal Report 10-186 (Submitted), ESAT-SISTA, K.U.Leuven (Leuven, Belgium), 2010, Lirias number: 270741.
- ▶ M. Signoretto, L. De Lathauwer and J. A. K. Suykens, A Kernel-based Framework to Tensorial Data Analysis, Internal Report 10-251 (Submitted), ESAT-SISTA, K.U.Leuven (Leuven, Belgium), 2010, Lirias number: 281359.
- ▶ M. Signoretto, L. De Lathauwer and J. A. K. Suykens, Convex Multilinear Estimation and Operatorial Representations, Internal Report 10-232, ESAT-SISTA, K.U.Leuven (Leuven, Belgium), 2010. NIPS Workshop: Tensors, Kernels and Machine Learning (TKML) 2010, Lirias number: 280419.
- ▶ M. Signoretto, L. De Lathauwer and J. A. K. Suykens, Kernel-based Learning from Infinite Dimensional 2-way Tensors, in ICANN 2010, part II, LNCS 6353, 2010, pp. 59-69., Lirias number: 270157.